

**Absolvovanie individuálnej  
odbornej praxe**

**Individual Professional Practice in  
the Company**

## Zadání bakalářské práce

Student: **Matúš Máčik**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R025 Informatika a výpočetní technika

Téma: **Absolvování individuální odborné praxe**  
**Individual Professional Practice in the Company**

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: Webtron, s.r.o.
2. Struktura závěrečné zprávy:
  - a) Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta
  - b) Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti
  - c) Zvolený postup řešení zadaných úkolů
  - d) Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe
  - e) Znalosti či dovednosti scházející studentovi v průběhu odborné praxe
  - f) Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vedl odbornou praxi studenta.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Mgr. Pavla Dráždilová, Ph.D.**

Konzultant bakalářské práce: Bc. Peter Štetiar

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014

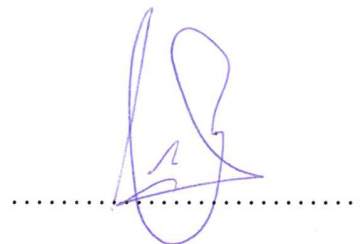


doc. Dr. Ing. Eduard Sojka  
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

V Ostrave 15. apríl 2014

A handwritten signature in blue ink is positioned over a horizontal dotted line. The signature is stylized, with a large loop at the top and a smaller loop at the bottom. The dotted line extends across the width of the signature.

Rád by som poďakoval pánovi Bc. Petrovi Štetiarovi za možnosť absolvovať odbornú prax vo firme Webtron s.r.o. a nadobudnúť skúsenosti a poznatky z práce na reálnych projektoch. Ďakujem aj Mgr. Pavle Dráždilovej za pomoc a vedenie pri vypracovávaní tejto odbornej práce.

## **Abstrakt**

Cieľom tejto bakalárskej práce je zhrnutie absolvovania individuálnej odbornej praxe. Stručne popisuje zameranie firmy, moje pracovné zaradenie a úlohy, ktoré boli počas tejto odbornej praxe zadané a vykonané. Hlavnou náplňou praxe bol redizajn internetovej služby [www.program.sk](http://www.program.sk) po grafickej, funkčnej a programovej stránke. Ďalšími úlohami boli Facebook aplikácie a Android aplikácia. V závere sa nachádza zhrnutie a zhodnotenie prínosu a výsledkov tejto odbornej praxe.

**Kľúčové slová:** HTML5, CSS3, jQuery, PHP, MySQL, web aplikácia, televízny program, Facebook aplikácia, Android aplikácia

## **Abstract**

The aim of this bachelor thesis is the summation of passing individual professional practice. It briefly describes the focus of the company, my job title and implemented tasks. Redesign of Internet service [www.program.sk](http://www.program.sk) graphically, functionally and programtically was the main scope of practice. Facebook applications and Android application were other tasks to solve. Summary and evaluation of the benefits and results from professional practice are in the end of this document.

**Keywords:** HTML5, CSS3, jQuery, PHP, MySQL, web application, television program, Facebook application, Android application

## **Zoznam použitých skratiek a symbolov**

PHP	– Hypertext Preprocessor
HTML	– HyperText Markup Language
CSS	– Cascading Style Sheets
SQL	– Structured Query Language
MVC	– Model-view-controll framework
SEO	– Search engine optimization
API	– Application programming interface
SDK	– Software development kit
AJAX	– Asynchronous JavaScript and XML
UX	– User experience
HTTPS	– Hypertext Transfer Protocol Secure

## Obsah

<b>1</b>	<b>Úvod</b>	<b>4</b>
<b>2</b>	<b>Zameranie firmy a pracovné zaradenie študenta</b>	<b>5</b>
2.1	Popis firmy Webtron s.r.o. . . . .	5
2.2	Pracovné zaradenie študenta . . . . .	5
2.3	Pracovné prostredie . . . . .	5
<b>3</b>	<b>Harmonogram práce</b>	<b>6</b>
3.1	Zhrnutie času stráveného nad jednotlivými úlohami či technológiami . . .	6
<b>4</b>	<b>Popis použitých technológií</b>	<b>8</b>
4.1	PHP . . . . .	8
4.2	MySQL . . . . .	8
4.3	HTML5 . . . . .	8
4.4	CSS3 . . . . .	8
4.5	jQuery . . . . .	9
4.6	Java . . . . .	9
<b>5</b>	<b>Zadané úlohy</b>	<b>10</b>
5.1	Program.sk . . . . .	10
5.2	Facebook aplikácie . . . . .	11
5.3	Android aplikácia . . . . .	11
<b>6</b>	<b>Postup riešenia jednotlivých úloh</b>	<b>12</b>
6.1	Program.sk . . . . .	12
6.2	Facebook aplikácie . . . . .	17
6.3	Android aplikácia . . . . .	21
<b>7</b>	<b>Záver</b>	<b>23</b>
<b>8</b>	<b>Reference</b>	<b>24</b>

## Seznam obrázků

1	Porovnanie dizajnu pre mobilné zariadenia so šírkou obrazovky do 320px v poradí: pôvodný stav, dodaný návrh a výsledné riešenie. . . . .	14
2	Facebook aplikácia Prezidentské voľby 2014 - výsledky po hlasovaní. . . .	20
3	Ukážka prispôsobenia webovej stránky na smartfóne s rozlíšením 480x800 pixelov. . . . .	21



## Zoznam výpisov zdrojového kódu

1	Ukážka media query v CSS3 . . . . .	13
2	MySQL dopyt na výpis vybraných televíznych staníc po polnoci. . . . .	16
3	Funkcia na overenie prihlásenia v PHP knižnici alebo prípadné presmerovanie na registráciu. . . . .	19
4	Ukážka overenia prihlásenia, uloženie informácií o užívateľovi a prípadné uloženie hlasu v ankete. . . . .	19
5	Odchytávanie možnej migrácie na inú web stránku. . . . .	22

## 1 Úvod

Táto bakalárska práca sa zaoberá študentskou odbornou praxou, ktorú som vykonával počas dvoch semestrov vo firme Webtron s.r.o. so sídlom v Žiline. Táto spoločnosť sa zaoberá vo väčšine tvorbou a rozvojom vlastných internetových služieb. Firmu som si vybral na základe predošlej spolupráce, kedy som na zákazku vypracovával grafické návrhy webových stránok. Prostredie a postupy vo firme mi teda boli známe a nemal som problém zaradiť sa do kolektívu. Hlavnou motiváciou pre absolvovanie študentskej odbornej praxe bolo dosiahnuť skúsenosti z práce na ucelených projektoch následne nasadených do reálnej prevádzky.

Moja práca spočívala vo vypracovaní postupne zadávaných úloh, analýze konkurenčných služieb, návrhu vlastného riešenia a zkomponovania požiadaviek zo strany firmy, celkovej implementácie, konzultáciách, doladovaní chýb a v konečnom nasadení do reálnej prevádzky. Na úlohach som prevažne pracoval sám. Všetky dôležité kroky som vždy konzultoval s mojím vedúcim, pánom Petrom Štetiarom.

V nasledujúcich kapitolách som sa zameral na popis priebehu celej praxe. Dôraz som kládol predovšetkým na projekt Program.sk, ktorý bol zo všetkých úloh najväčší. Tento projekt je internetová služba, ktorá užívateľom ponúka aktuálny televízny program vybraných televíznych staníc. K tejto službe som neskôr vytvoril aj Android aplikáciu pod rovnakým označením Program.sk. Určitý čas som strávil aj štúdiom a vytvorením Facebook ankiet. Zdôraznil som problémy, ktoré vyplývali počas práce na jednotlivých úlohách a ich riešenia. Nadobudnuté poznatky a prínos z praxe som zhrnul v závere.

## **2 Zameranie firmy a pracovné zaradenie študenta**

V tejto kapitole v krátkosti popisujem zameranie firmy, moje pracovné zaradenie a prostredie, v ktorom som vykonával odbornú študentskú prax.

### **2.1 Popis firmy Webtron s.r.o.**

Firma Webtron s.r.o. vznikla v roku 2009 ako internetová mediálna spoločnosť s cieľom vytvárať kvalitné internetové služby a reklamu pre svojich zákazníkov. Firma neskôr začala pracovať na vlastných projektoch, napríklad [www.dobre-recepty.sk](http://www.dobre-recepty.sk), [www.mapa-mapy.sk](http://www.mapa-mapy.sk), [www.info.sk](http://www.info.sk), [www.program.sk](http://www.program.sk) a ďalších. Tieto projekty časom získali bohatú návštevnosť presahujúcu 1 000 000 jedinečných návštev mesačne. Preto sa firma v súčasnosti orientuje na rozvoj a skvalitnenie vlastných projektov a týmto spôsobom chce zvýšiť ich profitabilitu a konkurencieschopnosť.

### **2.2 Pracovné zaradenie študenta**

Vo firme som bol zaradený na pozíciu programátora webových aplikácií, čo primárne zahŕňa programovanie v jazyku PHP a využitie databázy MySQL. Bolo mi však pridelených viac komplexnejších úloh, takže okrem programovania v spomenutých jazykoch som pracoval aj ako kóder v jazyku HTML5, CSS3. Pre naprogramovanie interaktívnych funkcií som používal Javascript za použitia knižnice jQuery a pre Android aplikáciu som použil Javu. Podrobnejší popis úloh, na ktorých som pracoval uvádzam v nasledujúcich kapitolách.

### **2.3 Pracovné prostredie**

Počas prvého semestra som prax vykonával vo vlastnej kancelárii neďaleko bydliska alebo priamo z domova, pretože firma sídli v 30 km vzdialenom meste. Do sídla firmy som dochádzal na konzultácie sporadicky, keďže väčšina konzultácií prebiehala prostredníctvom internetovej komunikácie alebo telefonicky. V druhom semestri som chodil do sídla firmy pravidelne každý deň, pretože som časom zistil, že moja efektivita práce v domácom prostredí rapídne klesá.

### 3 Harmonogram práce

Počas praxe som si viedol pracovný denník, na základe ktorého som v podkapitole 3.1 popísal čas strávený riešením jednotlivých úloh. Mnohé z nižšie uvedených úloh sa často kombinovali. Celkovo som vo firme strávil 55 dní. Čas strávený prácou vo voľnom čase neuvádzam.

Vytvoriť televízny program pod projektovým označením Program.sk sa na prvý pohľad zdal ako pomerne jednoduchá úloha. V konečnom dôsledku som však nad týmto projektom strávil 40 dní.

Druhou, pomerne jednoduchou úlohou bolo vytvorenie troch Facebook ankiet. Najviac času mi v tomto prípade zabralo štúdium Facebook SDK pre PHP [12] a pravidiel takýchto aplikácií [13].

Na záver som dostal za úlohu vytvoriť Android aplikáciu pod názvom Program.sk, slúžiacu na zobrazovanie televízneho programu. Potreboval som však na ňu viac času, pretože s programovacím jazykom Java a Android API [14] som sa stretol len na dvoch vyučovacích predmetoch, preto som viac času strávil štúdiom týchto technológií.

#### 3.1 Zhrnutie času stráveného nad jednotlivými úlohami či technológiami

Presné časové údaje neuvádzam. Čas strávený prácou s jednotlivými technológiami a úlohami zhrnul do dní, pretože denník neobsahoval rozpis práce po hodinách.

- Celkovo 5 dní som sa oboznamoval so zadanými úlohami a požiadavkami, preskúmaním konkurenčných služieb, štúdiom interného frameworku a rôznymi API ako Facebook SDK pre PHP, či Android API.
- 15 dní som strávil tvorbou HTML šablón pre televízny program a Facebook aplikácie. Z toho najviac času mi zabrala optimalizácia CSS štýlu pre responzívny dizajn a jeho testovanie. SEO, čiže optimalizáciu pre vyhľadávače som zakomponoval priamo do tvorby HTML šablón, aby som neskôr nemusel meniť štruktúru už vytvorených dokumentov.
- Samotné programovanie skriptov bežiacich na strane serveru mi trvalo zhruba 8 dní. Použitými technológiami boli PHP a MySQL, z ktorých som v prípade televízneho programu vytvoril kontroléry a modelovú vrstvu. Pri Facebook aplikáciach som pomocou nich vytvoril vlastnú knižnicu využívajúcu Facebook SDK pre PHP, ktorá zabezpečovala komunikáciu s Graph API [11] a obsluhovala databázu s výsledkami hlasovania užívateľov, nakoľko sa jednalo o ankety.
- Programovaním interaktívnych funkcií a AJAX dopytov v jQuery som strávil zhruba 8 dní.

- Celkové testovanie a doladovanie či už televízneho programu, Facebook aplikácií či samotnej Android aplikácie zabralo 10 dní. Tento proces zaíňal aj testovanie na rôznych operačných systémoch a zariadeniach s rôznym rozlíšením obrazovky.
- Štúdium jazyka Java, jeho použitie s Android API a implementácia konkrétneho riešenia mi zabrala približne 8 dní. Najviac času som v tomto prípade strávil hľadaním najlepšieho spôsobu implementácie jednotlivých funkcií a tried, pretože takmer vždy existuje viacero možných variant so síce rovnakým výsledkom, ale výpočetný čas a náročnosť sa môže líšiť.

## 4 Popis použitých technológií

V nasledujúcej časti som stručne popísal technológie, ktoré som počas praxe použil na riešenie jednotlivých úloh. Pri každej z nich v krátkosti uvádzam aj spôsob použitia danej technológie.

### 4.1 PHP

PHP (Hypertext Preprocessor) je open source skriptovací jazyk, ktorý sa používa najmä na programovanie klient-server aplikácií bežiacich na strane servera a pre vývoj dynamických webových stránok. PHP nie je kompilovaný, ale interpretovaný webovým serverom vo chvíli jeho volania a jeho výstupom môže byť HTML [1][2]. PHP som použil pri vytváraní modelovej vrstvy a kontrolérov televízneho programu a tiež pri tvorbe Facebook ankiet.

### 4.2 MySQL

MySQL je viacvláknový a viacužívateľský SQL relačný databázový server. Databáza je v MySQL tvorená z jednej a viac tabuliek, delených na riadky a stĺpce. V riadkoch sa nachádzajú jednotlivé záznamy, zatiaľ čo stĺpce určujú dátový typ jednotlivých záznamov. Práca s MySQL databázou je vykonávaná pomocou dopytov, ktoré vychádzajú z jazyka SQL[3][4]. Pomocou MySQL som zabezpečoval všetky úkony súvisiace s ukladaním a prácou s dátami v databáze bežiacej na serveri.

### 4.3 HTML5

HTML (HyperText Markup Language) je hypertextový značkovací jazyk. Zdrojový kód obsahuje takzvané tagy, ktoré označujú prvky obsahu a tým udávajú ich sémantický význam. HTML5 je nová generácia jazyka HTML, ktorá prináša množstvo nových elementov, no na druhej strane ich aj niekoľko vypustila [5]. Výsledkom je zjednodušenie práce pri písaní kódu a zvýšenie významu prvkov v takomto dokumente. Ďalším prínosom HTML5 je aj podpora multimédií bez nutnosti inštalovať rôzny podporný software ako Flash a podobne. Výraznou črtou HTML5 je aj rozhranie na vykresľovanie grafiky. Tento jazyk som využil pri všetkých výstupoch odosielaných pre klienta a následne zobrazovaných v internetovom prehliadači.

### 4.4 CSS3

Kaskádové štýly alebo CSS (Cascading Style Sheets) je všeobecné rozšírenie HTML. CSS je označované ako jednoduchý mechanizmus na vizuálne formátovanie internetových dokumentov. Vďaka CSS je možné oddeliť formu od obsahu, čo má za následok jednoduchý a prehľadný HTML kód [6]. Najnovšou verziou je CSS3, ktorá oproti predošlým verziám obsahuje množstvo nových vlastností [7]. Okrem bežných grafických nastavení ponúka aj vytváranie jednoduchých animácií. Zásadným problémom pri používaní CSS nastáva pri

jeho vykresľovaní v rôznych prehliadačoch, ktoré ho môžu interpretovať s odchylkami. Jediným možným riešením je optimalizovať CSS pre väčšie množstvo prehliadačov a pre niektoré prehliadače dokonca vytvoriť osobitné CSS súbory. CSS som používal súčasne s HTML, nakoľko výstupy pre klienta museli spĺňať predpoklady modernej internetovej aplikácie.

## 4.5 jQuery

jQuery je rýchla a presná cross-browser JavaScript knižnica, ktorá zjednodušuje v HTML dokumente udalosti, manipuláciu, animáciu a Ajax pre rýchly vývoj internetových, prevažne klientských aplikácií. Je to slobodný a otvorený software pod MIT licenciou. jQuery tiež poskytuje možnosti pre vývojárov na vytváranie pluginov postavených na tejto JavaScript knižnici [8]. Vďaka jQuery som naprogramoval interaktívne a v mnohých prípadoch asynchrónne, na pozadí bežiacie funkcie, ktoré aplikácie obohatili o rýchlosť a lepší dojem z používania aplikácii.

## 4.6 Java

Jazyk Java vznikol prebratím syntaxe jazyka C a objektového modelu z jazyka C++. Hlavným rysom tejto technológie je, že zdrojové kódy nie sú kompilované do strojového kódu, ale do takzvaného byte-kódu, ktorý nie je závislý od konkrétnej platformy [9]. To znamená, že je možné aplikácie spúšťať na rôznych procesoroch, mobiloch, či iných zariadeniach. Aby aplikácie napísané v tomto jazyku mohli byť spustené, je potrebné aby platforma kde aplikáciu spúšťame, obsahovala virtuálny stroj jazyka Java. Tento software v zariadení vystupuje ako hardwarové zariadenie, ktoré zabezpečuje preklad z byte-kódu do natívnych inštrukcií hostiteľského počítača [10]. Jazyk Java v spojení s Android API som použil pri programovaní Android aplikácie, ktorá zobrazovala televízny program.

## 5 Zadané úlohy

V tejto kapitole sa zameriam na popis jednotlivých úloh zadaných počas praxe a požiadaviek na ich implementáciu.

### 5.1 Program.sk

Prvou úlohou bolo vyriešiť redizajn internetovej stránky *www.program.sk* a vytvoriť z nej interaktívnu a užívateľsky jednoduchú webovú aplikáciu. Táto stránka ponúkala užívateľom prehľad televíznych programov vybraných televíznych staníc. Užívateľ si mohol preklikom zobrazovať detailné informácie o jednotlivých televíznych reláciách, listovať v programe po dňoch alebo zobrazovať výpis relácií od určitej hodiny. Redizajn sa netýkal iba vzhľadu, ale aj samotného programového jadra služby. Hlavnou podmienkou bolo prispôbenie zobrazenia pre väčšinu dnešných zariadení od telefónov a tabletov po širokouhlé monitory za použitia techniky nazývanej responzívny dizajn.

#### 5.1.1 Požiadavky na návrh a implementáciu

Nasledujúci zoznam obsahuje požiadavky na redizajn stránky *www.program.sk*, funkčné požiadavky a ďalšie požiadavky na celkovú implementáciu.

- Vizuálna stránka projektu bude nakódovaná v HTML5 a CSS3.
- Vizuálna stránka projektu bude obsahovať responsive design.
- Jadro projektu bude naprogramované nad vlastným MVC frameworkom firmy.
- Minimalizácia dotazov na server a množstva prenesených dát.
- Minimalizácia času spracovania PHP skriptov.
- Zvýšenie interaktivity užívateľskej časti.
- Výber a uloženie televíznych staníc užívateľom s možnosťou odoslania nastavení na e-mail a možnosť zdieľania nastavení medzi rôznymi zariadeniami.
- Viac možností zobrazenia televízneho programu implementované v jednom HTML kóde meniace sa na základe CSS tried bez znovunačítania stránky.
- Zobrazovanie detailu programu vo vyskakovacom okne.
- Zobrazovanie programu v rôznych časoch bez znovunačítania stránky.



## 5.2 Facebook aplikácie

Ďalšou úlohou, ktorú som dostal, bolo vytvoriť tri Facebook aplikácie slúžiace ako ankety. Prvá anketa bola zameraná na hlasovanie o výsledku prezidentských volieb 2014 na Slovensku, druhá na hlasovanie o tom ako dopadnú slovenskí hokejisti na ZOH v Sochi 2014 a posledná predstavovala opäť prieskum volebnej mienky v druhom kole prezidentských volieb. Aplikácie boli následne propagované na informačnom portáli [www.info.sk](http://www.info.sk) a na fanúšikovských stránkach spojených s týmto portálom. Dizajn aj logiku aplikácií som mal vytvoriť samostatne.

## 5.3 Android aplikácia

Posledným zadáním bolo naprogramovať aplikáciu bežiacu v operačnom systéme Android. Aplikácia s názvom Program.sk mala byť len internetovým prehliadačom, nakoľko webová aplikácia [www.program.sk](http://www.program.sk) bola navrhnutá a vypracovaná tak, aby sa správne zobrazovala pri akejkoľvek rozlíšení, čo znížilo náklady na vývoj tejto aplikácie na minimum.

### 5.3.1 Požiadavky na návrh a implementáciu

V tomto prípade som neobdržal mnoho požiadaviek, keďže aplikácia bola navrhnutá na veľmi jednoduchej báze.

- Javascript, CSS a obrázky budú uložené priamo v aplikácii.
- Po kliknutí na reklamu sa bude zobrazovať v primárnom internetovom prehliadači.

## 6 Postup riešenia jednotlivých úloh

Na nasledujúcich stranách detailne popisujem postup riešenia jednotlivých úloh. Najväčší dôraz kladiem na popis tvorby internetovej aplikácie slúžiacej na zobrazovanie televízneho programu, pretože to bola počas praxe časovo najnáročnejšia úloha.

### 6.1 Program.sk

Redizajn webovej stránky [www.program.sk](http://www.program.sk) slúžiacej na zobrazovanie televízneho programu spočíval v kompletnom grafickom redizajne, preprogramovaní jadra stránky a celkovom zatraktívnení tejto služby. Proces riešenia tejto úlohy začal analýzou pôvodného stavu. Z nej vyplynuli mnohé nedostatky, z ktorých som vypracoval návrhy na zmenu do novej verzie tejto služby. Hoci som obdržal grafické návrhy na novú verziu, kompletne som ich zmenil a vypracoval som svoju vlastnú verziu, viď podkapitola 6.1.2 . UX, inak povedané užívateľský dojem, som zvýšil pomocou interaktívnych funkcií naprogramovaných v jQuery, o čom píšem v podkapitole 6.1.3. Nakoniec stručne popisujem pozadie aplikácie bežiace na strane serveru v podkapitole 6.1.4.

#### 6.1.1 Analýza pôvodného stavu

Služba Program.sk fungujúca zhruba rok, bola klasickým výpisom časového harmonogramu vysielania vybraných televíznych staníc. Pôvodné riešenie obsahovalo možnosť zobrazíť výpis podľa dňa, zobrazenie výpisu od vybranej hodiny pre daný deň, zobrazenie týždenného výpisu pre vybranú stanicu a zobrazenie detailu jednotlivých relácií. Interakcia na strane klienta za použitia Javascriptu bola prezentovaná len možnosťou výpisu časového harmonogramu od vybranej hodiny, kedy nedochádzalo k znovunačítaniu stránky. Po vyhľadávaní kľúčovej frázy "tv program" bola stránka v [www.google.sk](http://www.google.sk) na 6. pozícii. Návštevnosť pred spustením novej verzie bola podľa dát z Google Analytics priemerne do 3000 návštev denne. Grafické spracovanie nezodpovedalo súčasným trendom vo svete web dizajnu.

#### 6.1.2 Grafický dizajn

Dodaný grafický návrh obsahoval náhľady pre rozlíšenia obrazoviek so šírkou do 320px, 768px, 979px a jednotný formát pre rozlíšenia nad 979px. Ako človeku so skúsenosťami v oblasti grafického návrhu web aplikácií sa mi niektoré riešenia uloženia prvkov zdali ne-logické, čitateľnosť textových informácií bola znížená a prvky klikateľné na smartfónoch a tabletoch neboli dostatočne veľké, aby ponúkli jednoduché ovládanie a znížili možnosť omylu prekliku na iný prvok.

Najväčšie pripomienky som mal k zvláštnemu rozloženiu rozklikávacieho menu pre nastavenie TV staníc, ktoré bolo pri každom rozlíšení rôzne umiestnené a naformátované. Navrhol som variantu takzvaného "modal window" alebo vyskakovacieho okna, keďže jeho prispôbenie pre rôzne rozlíšenia nie je obtiažne.

Ďalšou z pripomienok bola modernizácia grafického návrhu, pretože pôvodný návrh nepôsobil ako moderná web aplikácia, ale ako klasická web stránka.

Zvýšenie čitateľnosti textových informácií bol taktiež jedným z návrhov, ktoré by mohli byť realizované pomocou vhodne zvoleného riadkovania, veľkosti písma a horizontálnych a vertikálnych odstupov od ostatných elementov.

**Poznámka 6.1** V súčasnosti je v komunite pohybujúcej sa okolo tvorby web stránok ako web aplikácia chápaná stránka, ktorá ponúka interaktívne funkcie, moderný dizajn a dáta prispôbené užívateľovi. Avšak rozdiel medzi webovou stránkou a webovou aplikáciou nie je doposiaľ pevne stanovený a môžeme ho chápať ako subjektívny pohľad každej osoby. Tejto problematike sa venuje množstvo diskusií a článkov na internete [15].

Po podaní mojich pripomienok som dostal voľnú ruku pri návrhu a kódovaní vizuálnej stránky, a tak som mohol zapracovať všetky moje návrhy a dizajn stránok prispôbiť podľa mojich skúseností (viď obrázok 1).

Webová aplikácia Program.sk mala podľa požiadaviek spĺňať predpoklady responzívneho dizajnu. Táto forma grafickej prezentácie webovej aplikácie zabezpečuje dobrú ovládateľnosť a čitateľnosť textovej informácie bez potreby používať približovanie na obrazovkách s menším rozlíšením. Podľa predošlých štatistík nazbieraných pomocou Google Analytics sme mohli zhodnotiť, že je potrebné dizajn navrhnuť s maximálnou prispôboivosťou pre zhruba 800 a viac rôznych rozlíšení. Táto forma dizajnu sa vyznačuje jedným HTML kódom spoločným pre všetky zariadenia, pričom zmeny vzhľadu sú záležitosťou CSS. Pri návrhu responzívneho dizajnu sa používajú takzvané body zlomu, kedy sa dizajn stránky výraznejšie mení v závislosti na šírke obrazovky užívateľa. Túto vlastnosť je možné dosiahnuť prispôbením CSS vďaka podmienke nazývanej media query (viď výpis 1). Responzívny dizajn okrem jeho estetickej a funkčnej vlastnosti zjednodušuje vývoj internetových aplikácií, pretože jeho použitím a optimalizáciou vývojári nie sú nútení vytvárať a spravovať niekoľko verzií aplikácie pre rôzne druhy zariadení.

V návrhu som určil nasledujúce body zlomu reprezentujúce šírku obrazovky v pixeloch:

- viac ako 980px pre klasické monitory a zariadenia s väčším rozlíšením,
- 768px - 979px pre tablety a smartfóny,
- 480px - 767px smartfóny,
- menej ako 479px - mobilné telefóny a staršie smartfóny.

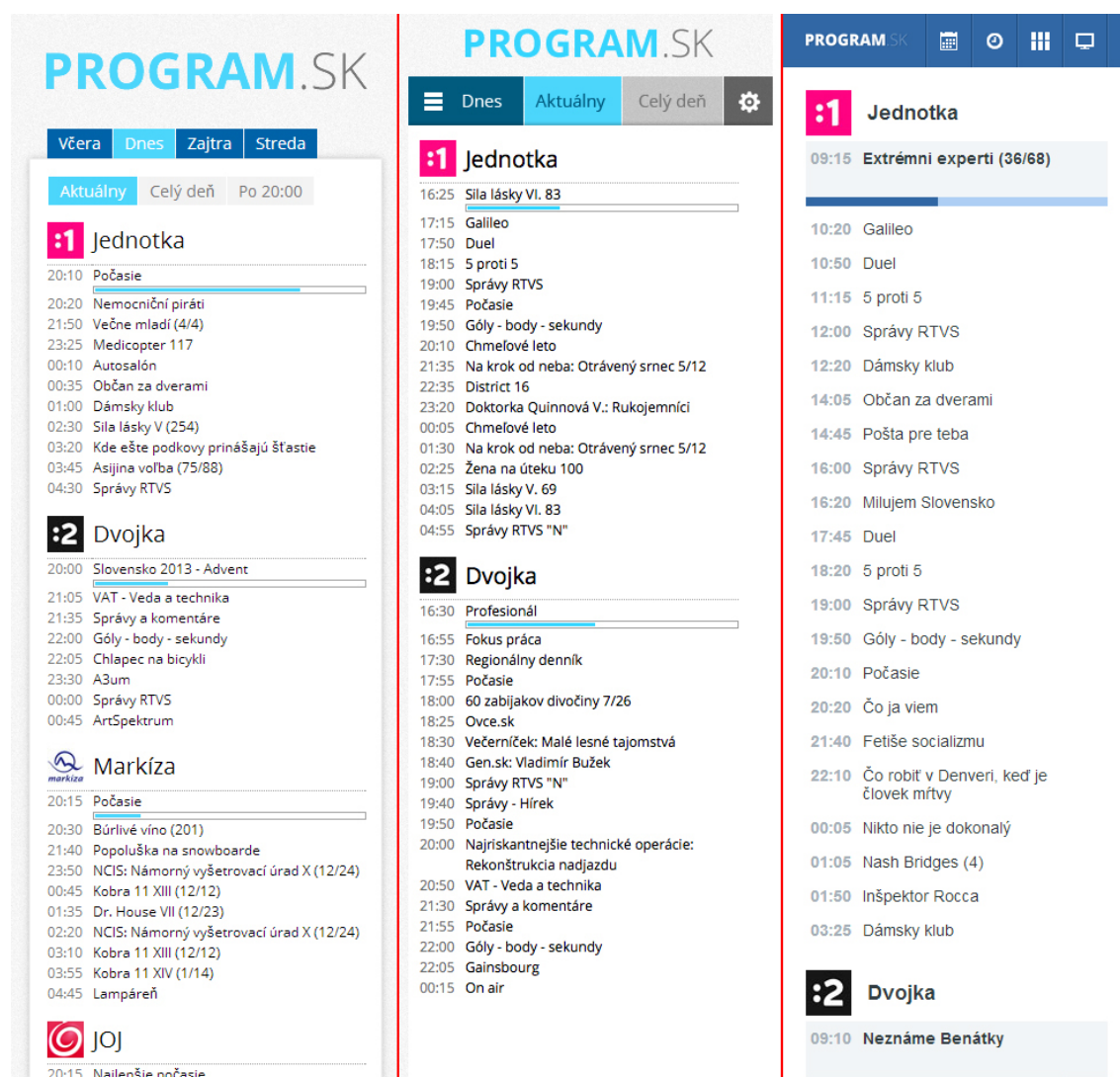
---

```
@media (max-width: 979px) {
  body{
    font-size:14px;
  }
}
```

---

Výpis 1: Ukážka media query v CSS3

Aby som zabezpečil maximálnu prispôsobivosť, od rozlíšenia pod 767px bola stránka takzvané plávajúca, čo je vlastnosť, kedy sa jednotlivé elementy prispôbujú obrazovke na základe percentuálnych pomerov.



Obrázek 1: Porovnanie dizajnu pre mobilné zariadenia so šírkou obrazovky do 320px v poradí: pôvodný stav, dodaný návrh a výsledné riešenie.

### 6.1.3 Interaktívne funkcie

Implementáciu interaktívnych užívateľských funkcií som zabezpečil pomocou knižnice jQuery. Vďaka týmto funkciám užívateľ nemusel čakať na znovunačítanie stránky a preklikávať sa medzi podstránkami, ale mal všetky údaje k dispozícii na hlavnej stránke. Touto úpravou sa šetril nielen čas užívateľa, ale aj výpočetný čas a takisto aj objem prenesených dát.

Zmena pohľadu výpisu dát prebieha čisto na strane klienta. HTML kód je optimalizovaný do takej úrovne, aby nemusela byť jeho štruktúra nijako zásadne menená pomocou jQuery. Zmena vzhľadu prebieha zámenou CSS identifikátorov, ktoré popisujú formu jednotlivých zobrazení. V ponuke je 5 rôznych výpisov, aby si každý užívateľ mohol vybrať podľa svojich preferencií. V prípade tejto interakcie dochádza aj k asynchrónnej komunikácii so serverom, kde je pre potreby štatistiky uložený údaj o type zobrazenia a späťne je klientovi odoslaný súbor cookie pre ďalšiu identifikáciu zariadenia pri ďalšej návšteve alebo znovunačítaní stránky.

Detail vybranej relácie, nastavenia a chybové či oznamovacie hlásenia sa zobrazujú vo vyskakovacom HTML okne. Ako som už spomenul, vďaka tomuto riešeniu dochádza k zníženiu objemu prenesených dát, nakoľko stránka sa neobnovuje, nenačítava sa celý dokument a tiež sa znižuje počet dopytov na server pre overovanie zmeny externých súborov ako obrázky, CSS súbory a podobne. V tomto prípade sa po kliknutí na objekt, ktorého obsah sa má zobraziť vo vyskakovacom okne, vytvorí asynchrónny jQuery dopyt na požadovaný obsah. PHP skript na strane serveru prijme požiadavku s potrebnými parametrami a po vyhodnotení posiela späťne vygenerovanú HTML štruktúru vyskakovacieho okna, ktoré je následne pomocou jQuery vložené a zobrazené v aktuálnom HTML dokumente.

V prípade nastavení si užívateľ môže zakliknúť svoje obľúbené televízne stanice a tiež určiť aj ich poradie vo výpise. Po uložení opäť jQuery posiela dopyt na server s vybranými stanicami. PHP vygeneruje pre nového užívateľa cookie s hodnotou obsahujúcou 32 znakov tvorených kombináciou malých písmen abecedy a čísel. Tento identifikátor je uložený v databáze a je k nemu priradený zoznam staníc v určenom poradí. V prípade užívateľa s už vygenerovanou cookie je len zmenený záznam v databáze. Súčasne prebehne aj obnova stránky, aby došlo k načítaniu nových nastavení. Užívateľovi je vygenerovaná jedinečná URL adresa, pomocou ktorej si môže svoje nastavenia otvoriť v ľubovoľne zvolenom zariadení. Zmena nastavení v akomkoľvek z týchto zariadení sa preukáže súčasne na všetkých zariadeniach.

### 6.1.4 Serverová časť aplikácie

Pozadie celej aplikácie som programoval v jazyku PHP za použitia MySQL databázy. Aplikácia bola vytvorená nad interným MVC frameworkom firmy. Do tohto frameworku som prístup nemal, nakoľko ho aktuálne využívalo niekoľko web stránok. Na serveri, kde bežal framework, mi bola vytvorená testovacia zložka s FTP prístupom, kde som mohol pracovať na vývoji. Mojou úlohou bolo vytvoriť HTML pohľady, modelovú vrstvu a kontrolery, ktoré aplikáciu obsluhujú. Po naštudovaní si samotného frameworku som začal s úpravou pôvodných tabuliek v databáze a vytvorením nových tabuliek pre uloženie nastavení. Samotné kontrolery v podstate veľa práce nevykonávajú. Prijímajú HTTP dopyty na základe ktorých načítavajú dáta z modelu a následne tieto dáta vkladajú na určené miesta v pohľadoch. Kontrolery overujú tiež všetky vstupy z formulárov, prípadne parametre v URL adresách, aby nedošlo k SQL injection.

Najzložitejším problémom bol výpis programu po polnoci. V prípade televízií sa vysielač deň chápe v priemere ako čas medzi 5:20 až 3:30 ďalšieho dňa. Po polnoci je síce už ďalší deň, stále sa však uvádza dátum predošlého. Mojou úlohou bolo po polnoci vypísať program tak, aby obsahoval relácie, ktoré začali pred polnocou a ešte stále trvajú, pripojiť k nemu zvyšné relácie príslušného dňa a následne pripojiť aj relácie nasledujúceho vysielacieho dňa. Tento problém som vyriešil v dvoch fázach. V prvej som si pomocou MySQL dopytu (viď výpis 2) vytiahol všetky relácie, ktoré začali po 21:00 daného dňa a k nim som pripojil celý nasledujúci deň. V druhej fáze som pomocou PHP postupne prechádzal relácie, až pokiaľ som nenašiel aktuálne bežiacu a všetky predošlé som následne zmazal.

---

```
SELECT * FROM( (SELECT id_program, day, id_station, title, start, end, length, order_num FROM
    $this->tbl_program WHERE day = $start_day AND order_num >= 21 AND id_station IN (
        $stations) AND active='a' ORDER BY order_num ASC) UNION ALL (SELECT id_program, day,
        id_station, title, start, end, length,order_num FROM $this->tbl_program WHERE day =
        $end_day AND id_station IN ($stations) AND active='a' ORDER BY order_num ASC) ) a
ORDER BY field(id_station, $stations), day, order_num;
```

---

Výpis 2: MySQL dopyt na výpis vybraných televíznych staníc po polnoci.

## 6.2 Facebook aplikácie

Pri vytváraní aplikácií pre prostredie Facebooku som dostal opäť voľnú ruku, nakoľko firma doposiaľ s touto oblasťou nepracovala a išlo v podstate o prvé oboznamovanie sa s tvorbou a testovaním aplikácií pre sociálnu sieť.

### 6.2.1 Čo je to Facebook aplikácia?

Facebook aplikácie primárne slúžia na zbieranie nových fanúšikov, budovanie značky, ale aj pre zábavu v podobe prevažne flash hier. Tieto aplikácie sa nenachádzajú priamo na Facebooku, ale na vlastnom hostingu ich majiteľa. To v podstate znamená, že samotná aplikácia môže byť naprogramovaná v akomkoľvek jazyku, ktorý má HTML alebo flash výstup. V tomto prípade sociálna sieť slúži iba ako prezentačné miesto a umožňuje takéto aplikácie načítavať do vnoreného okna v stránke. Aplikácia môže používať prístup k Graph API pomocou rôznych programovacích jazykov pre interakciu a získavanie informácií o užívateľoch.

Aplikácie je možné na Facebook implementovať pomocou dvoch možných platforiem.

- Canvas - je priestor pre nasadenie aplikácie s primárnou šírkou 760px, môže byť však aj plávajúci so zarovnaním vľavo. Táto platforma je vhodná predovšetkým pre hry, nakoľko aplikácia získa vlastnú Facebook URL adresu a nie je možné ju inštalovať do fanúšikovských stránok ako záložky. Canvas tiež umožňuje užívateľom pozývať priateľov priamo z aplikácie.
- Page Tab - ponúka šírku priestoru pre aplikáciu buď 520px alebo 810px. Je vhodná pre aplikácie typu súťaž, anketa, informačná stránka, e-shop a podobne. Tento typ je možné inštalovať do fanúšikovských stránok ako záložky a často sú označované aj ako "like-gate", čiže brána pre zbieranie fanúšikov.

### 6.2.2 Tvorba aplikácie

Prvým krokom pri riešení tohto zadania bolo naštudovanie si Graph API a možností, ktoré toto rozhranie ponúka. Následne som sa oboznámil s knižnicou Facebook SDK pre PHP, vďaka ktorej je možné pomocou PHP pristupovať do API. Ďalej všetky potrebné náležitosti pre správny chod aplikácie či rôznych náležitostí ako právna stránka veci alebo tvorba prezentačných obrázkov.

Po naštudovaní dokumentácie som sa rozhodol pre Page Tab aplikáciu kvôli možnosti inštalovať ju do záložiek na fanúšikovských stránkach. Keďže som mal za úlohu vytvoriť tri ankety a v budúcnosti počítalo sa s ďalšími, rozhodol som sa naprogramovať vlastnú PHP knižnicu z dôvodu opätovného použitia. Obsahovala potrebné funkcie a všeobecnú modelovú vrstvu a celá sa správala podľa konfiguračného súboru danej aplikácie. Táto knižnica využíva na pripojenie sa k Graph API rozhranie pod názvom Facebook SDK pre PHP. Bolo potrebné zabezpečiť aj SSL licenciu, nakoľko Facebook vyžaduje zabezpečené

pripojenie cez HTTPS k hostingu, kde sa aplikácia nachádza.

Aplikácie som programoval v PHP s použitím MySQL databázy, HTML, CSS a Javascript. Každá z nich používala moju PHP knižnicu na zistenie, či je užívateľ prihlásený do aplikácie (viď výpis 3). V prípade že nebol, bolo mu ponúknuté prihlásenie, po ktorom mu bolo sprístupnené hlasovanie a mne umožnilo zisťovať, či je fanúšikom vybraných Facebook stránok a poskytlo mi jeho osobné údaje (viď výpis 4).

Prvou anketou bol prezidentský volebný prieskum (viď obrázok 2), kde som od užívateľa požadoval prihlásenie sa do aplikácie pomocou Facebooku, nakoľko som potreboval jeho identifikačné údaje a súhlas s ich použitím, aby nedošlo k viacnásobnému hlasovaniu. Tento prístup sa ukázal ako nie veľmi príťažlivý. Ľudia si začínajú uvedomovať dôležitosť ochrany svojich údajov, čo dokazovali aj štatistiky z ktorých vyplývalo, že do aplikácie sa prihlásilo zhruba 5,5% z tých užívateľov, ktorí klikli na anketu, čo predstavuje len 361 užívateľov z celkového počtu 505.

Pri druhej ankete o tom, ako dopadnú slovenskí hokejisti na ZOH v Sochi 2014 som nevyžadoval od užívateľa žiadne informácie a prístup do aplikácie bol bez prihlásenia. Odhlasovaného užívateľa som identifikoval pomocou použitia cookie súborov. Táto anketa nebola veľmi atraktívna, preto sa neskôr aj prestala promovať. Celkovo v nej hlasovalo 98 ľudí.

Tretia anketa bola zameraná na volebný prieskum v druhom kole prezidentských volieb. Tentokrát som od užívateľov nevyžadoval žiadne prihlásenie ani údaje potrebné pre hlasovanie, aby sa zvýšil počet hlasujúcich. Počas prvých 3 dní hlasovalo takmer 1000 ľudí. Rozpoznávanie odhlasovaného užívateľa prebiehalo taktiež na základe cookie súborov.



---

```

public function isUserLogged(){

    if ($this->getUser()) {
        try {
            $this->user_data = $this->api('/me','GET');
            return true;
        } catch(FacebookApiException $e) {}
    }
    // login redirect
    $params = array(
        'redirect_uri' => $this->FB_APP_URL,
        'scope' => $this->PERMISSIONS
    );
    echo '<script>top.location="" . $this->getLoginUrl($params) . "";</script>';
    exit();
}

```

---

Výpis 3: Funkcia na overenie prihlásenia v PHP knižnici alebo prípadné presmerovanie na registráciu.

---

```

require(' ../ facebook_core/fb_app_core.php');
$fb = new FB_app_core();

if ($a->isUserLogged()){

    $user = $a->getUserData();
    $output['name'] = $user['first_name'];

    if (!$fb->isUserInDatabase()){
        $fb->addUserInToDatabase();
    }

    $output['has_voted'] = $fb->hasUserVoted();

    if (!$output['has_voted'] && $_GET['vote'] && array_key_exists($_GET['vote'],$fb->objects)){

        $output['vote'] = $_GET['vote'];

        if ( $fb->addVote($_GET['vote']) ){
            $output['has_voted'] = true;
        }

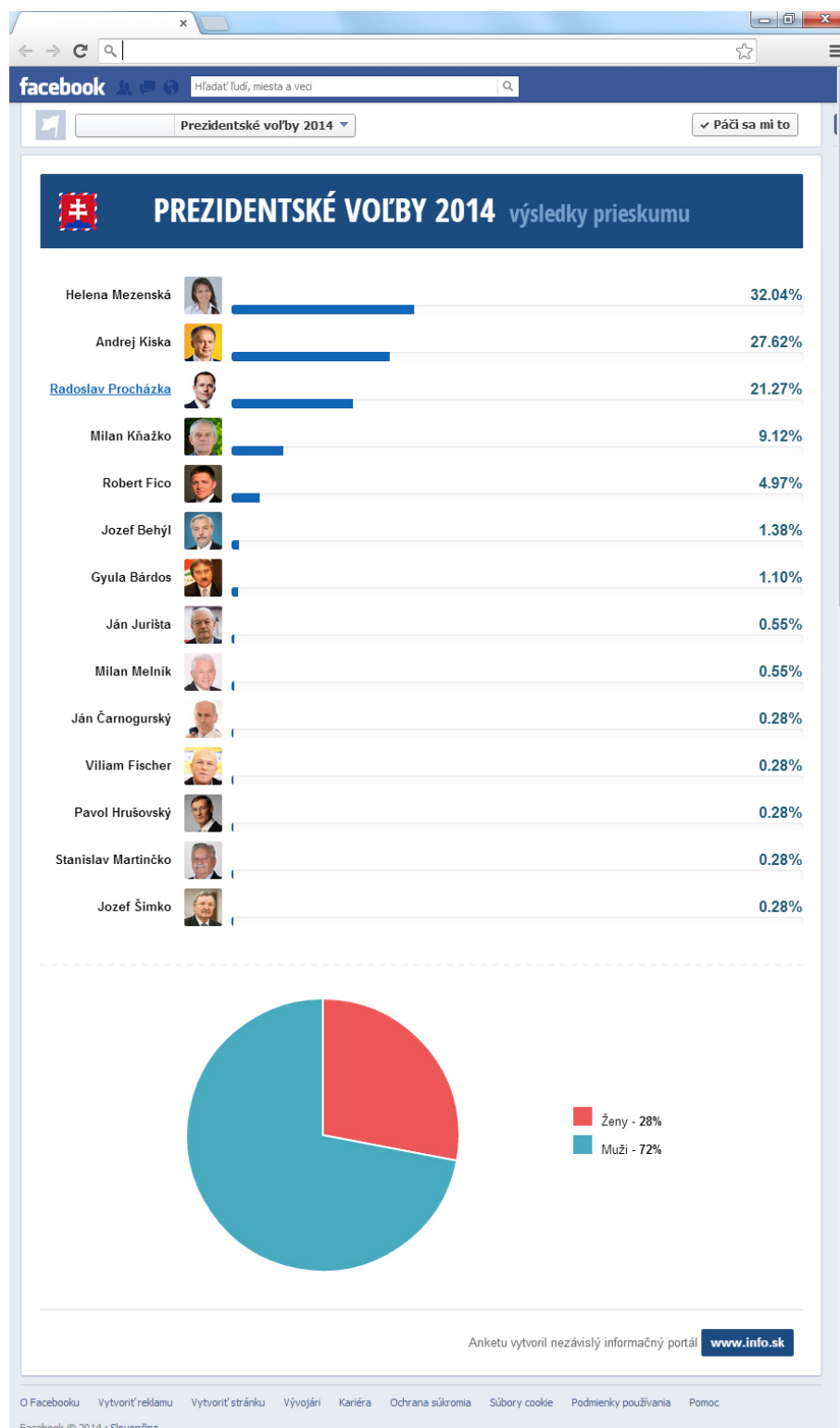
    }

}

```

---

Výpis 4: Ukážka overenia prihlásenia, uloženie informácií o užívateľovi a prípadné uloženie hlasu v ankete.



Obrázek 2: Facebook aplikácia Prezidentské voľby 2014 - výsledky po hlasovaní.

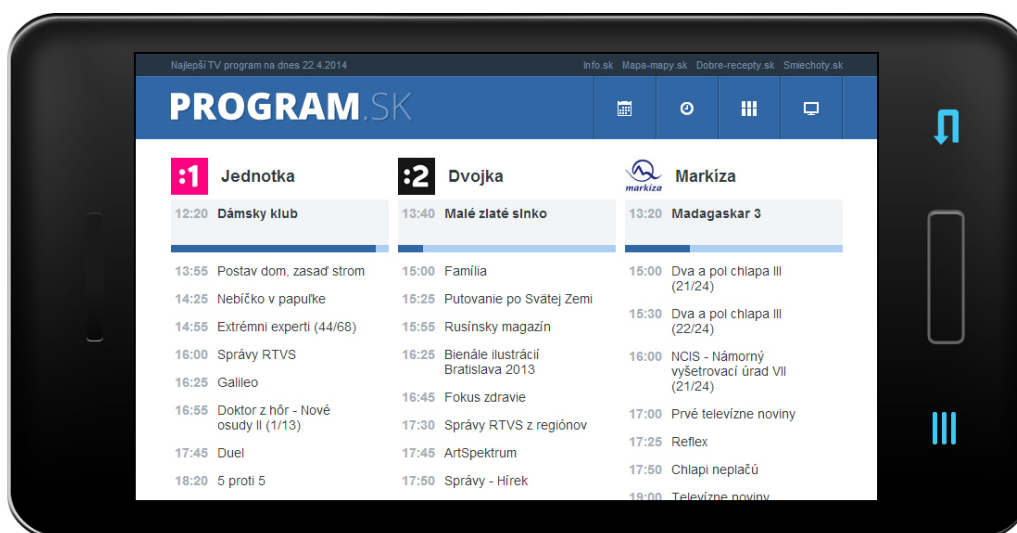
## 6.3 Android aplikácia

Naprogramovanie aplikácie s názvom Program.sk bežiackej na operačnom systéme Android bola pomerne jednoduchá úloha. Aplikácia bola rozšírením projektu Program.sk a mala takisto zobrazovať televízny program. Podľa zadania aplikácia mala slúžiť ako internetový prehliadač, nakoľko internetové stránky projektu Program.sk sú optimalizované aj pre zariadenia s malým rozlíšením (obrázok 3). Tento fakt ušetril náklady na vývoj samotnej aplikácie aspoň do doby, kedy podľa štatistík bude rozumné investovať do vývoja sofistikovanejšej podoby aplikácie.

### 6.3.1 Tvorba aplikácie Program.sk

Pri programovaní Android aplikácie som musel ošetriť množstvo výnimiek, ktoré mohli nastať a ohroziť správne fungovanie aplikácie počas práce.

Keďže aplikácia komunikuje so serverom a prijíma dáta, je potrebné zabezpečiť kontrolu aktívneho pripojenia k internetu. Pre túto potrebu som vytvoril triedu s podtriedou, ktorá asynchrónne s ohľadom na hlavné vlákno aplikácie overí internetové pripojenie. Overovanie prebieha v dvoch fázach. V prvej za pomoci knižnice ConnectivityManager zisťuje, či je na zariadení aktívne aspoň jedno z možných sieťových pripojení ako Wi-Fi, GPRS, UMTS, atď. V prípade, že existuje aktívne sieťové pripojenie, trieda sa pomocou dopytu na určenú URL adresu pokúsi napojiť na server a čaká na odpoveď. V prípade odpovede s kódom 200 trieda vracia kladnú odpoveď a dá sa predpokladať, že existuje aj pripojenie k internetu a aplikácia je schopná načítať obsah web stránky Program.sk. V prípade, že nie, zobrazí sa cez celú obrazovku chybová hláška.



Obrázok 3: Ukážka prispôsobenia webovej stránky na smartfóne s rozlíšením 480x800 pixelov.

Ak existuje internetové pripojenie, aplikácia sa pokúsi pomocou ďalšej asynchrónnej triedy načítať HTML dokument televízneho programu ako textový reťazec. Keď je HTML uložené v premennej, hlavná trieda aplikácie načíta túto premennú do WebView, čo je komponent slúžiaci ako prehliadač postavený na jadre WebKit. Načítavanie do prehliadača prebieha pomocou funkcie `loadDataWithBaseURL()`, ktorá zmení všetky relatívne URL adresy k CSS, Javascriptu a k obrázkom na absolútne s cestou odkazujúcou na adresár v aplikácii. Týmto spôsobom z neho načíta potrebné súbory. V porovnaní s verziou pre bežné prehliadače HTML stránky museli byť pozmenené a všetky relatívne cesty odkazov na podstránky prepísané na absolútne. V inom prípade by odkazy tiež smerovali do adresára v aplikácii.

Jednou z funkčných požiadaviek na aplikáciu bolo otváranie reklám v predvolenom internetovom prehliadači, čo som zabezpečil pomocou `WebViewClient` knižnice. Táto knižnica dokáže prijímať požiadavky od užívateľa, čiže jeho kliknutia a prácu s nimi. Preto je URL každého kliku overená (viď výpis 5), či náhodou nesmeruje na inú web stránku ako je `www.program.sk`. Ak áno, daná URL je otvorená v niektorom z prehliadačov pomocou takzvaného `Intent` objektu.

---

```
browser = (WebView)findViewById(R.id.webView1);
browser.setWebViewClient(new WebViewClient() {

    @Override
    public boolean shouldOverrideUrlLoading(WebView view, String url) {
        if (url.startsWith("http://www.program.sk")) {
            view.loadUrl(url + "?m=android");
            return true;
        }
        Intent browserIntent = new Intent(Intent.ACTION_VIEW, Uri.parse(url));
        startActivity(browserIntent);
        return false;
    }
});
```

---

Výpis 5: Odchytávanie nožnej migrácie na inú web stránku.

## 7 Záver

V priebehu praxe som využil množstvo poznatkov, s ktorými som sa oboznámil a stretol aj počas štúdia. Tieto vedomosti by však bez ďalšej vlastnej iniciatívy neboli dostačujúce. Väčšinu poznatkov som nadobudol vlastným úsilím počas celého štúdia a tiež počas vykonávania odbornej praxe.

Predmety, ktoré som počas štúdia absolvoval a mohol využiť počas praxe boli:

- Úvod do databázových systémů (UDBS),
- Databázové a informační systémy (DAIS),
- Programovací jazyky (PJ I),
- Vývoj internetových aplikací (VIA),
- Úvod do softwarového inženýrství (SWI),
- Tvorba aplikací pro mobilní zařízení II (TAMZ II).

Z uvedených predmetov boli pre mňa najprínosnejšie Programovací jazyky I a Tvorba aplikací pro mobilní zařízení II kde som sa oboznámil s jazykom Java a tiež jeho použitím na programovanie aplikácií pre operačný systém Android. Úroveň teoretických vedomostí nadobudnutých počas štúdia bola v dostačujúcej kvalite, avšak kôli praxi som si musel mnoho ďalších doštudovať. Samoštúdiom som nadobudol vedomosti hlavne o PHP, MySQL, HTML, CSS, Javascript a jQuery.

Absolvovanie odbornej praxe bolo pre mňa veľkým prínosom, nakoľko som sa naučil množstvo nových poznatkov a postupov pri programovaní nielen internetových, ale aj Android aplikácií. V mnohých prípadoch som otestoval aj svoju psychiku a vytrvalosť pri riešení niektorých problémov a ich viacnásobnom refaktorovaní. Vďaka riešeniu reálnych problémov som nadobudol skúsenosti, ktoré by som pri vypracovávaní teoretickej bakalárskej práce pravdepodobne nemohol získať.

Matúš Máčik

## 8 Reference

- [1] *PHP Manual* [online]. 2001 - 2014 [cit. 2014-04-01]. Dostupné z: <http://www.php.net/manual/en/>
- [2] GUTMANS, Andi. *Mistrovství v PHP 5*. Vyd. 2. Brno: Computer Press, 2007, 655 s. ISBN 978-80-251-1519-0
- [3] KOFLER, Michael. *Mistrovství v MySQL 5*. Vyd. 1. Překlad Jan Svoboda, Ondřej Baše, Jaroslav Černý. Brno: Computer Press, 2007, 805 s. ISBN 978-80-251-1502-2
- [4] *MySQL Documentation* [online]. 1997 - 2014 [cit. 2014-04-01]. Dostupné z: <http://dev.mysql.com/doc/>
- [5] HTML Reference - (HTML5 Compliant). *W3Schools* [online]. 1999 - 2014 [cit. 2014-04-01]. Dostupné z: <http://www.w3schools.com/tags/>
- [6] CROFT, Jeff, Ian LLOYD a Dan RUBIN. *Mistrovství v CSS: pokročilé techniky pro webové designéry a vývojáře*. Vyd. 1. Překlad Josef Bábík. Brno: Computer Press, 2007, 409 s. ISBN 978-80-251-1705-7
- [7] CSS Reference. *W3Schools* [online]. 1999 - 2014 [cit. 2014-04-01]. Dostupné z: <http://www.w3schools.com/cssref/default.asp>
- [8] *jQuery Core API Documentation* [online]. 2014 [cit. 2014-04-01]. Dostupné z: <http://jquery.com/>
- [9] SCHILDT, Herbert. *Java 7: výukový kurz*. 1. vyd. Brno: Computer Press, 2012, 664 s. ISBN 978-80-251-3748-2
- [10] SPELL, Brett. *Java: programujeme profesionálně*. Vyd. 1. Překlad Bogdan Kiszka. Praha: Computer Press, 2002, 1022 s. Programujeme profesionálně. ISBN 80-722-6667-5
- [11] *Graph API documentation* [online]. 2014 [cit. 2014-04-01]. Dostupné z: <https://developers.facebook.com/docs/graph-api/>
- [12] *Facebook SDK for PHP* [online]. 2014 [cit. 2014-04-01]. Dostupné z: <https://developers.facebook.com/docs/reference/php>
- [13] *Facebook Platform Policy* [online]. 2014 [cit. 2014-04-01]. Dostupné z: <https://developers.facebook.com/policy/>
- [14] *Introduction to Android* [online]. 2014 [cit. 2014-04-01]. Dostupné z: <https://developer.android.com/guide/index.html>
- [15] *Odkazy na diskusiu a článok o rozdiel medzi web aplikáciou a web stránkou*  
<http://stackoverflow.com/questions/8694922/whats-the-difference-between-a-web-site-and-a-web-application>  
<http://www.visionmobile.com/blog/2013/07/web-sites-vs-web-apps-what-the-experts-think/>